AD-A120 351 TUSKEGEE INST AL SCHOOL OF ENGINEERING F/G 15/5
THREE COMPUTER BASED AIDS TO MAINTENANCE SCHEDULING.(U)
SEP 82 A HARGROVE AFOSR-81-0168

UNCLASSIFIED AFOSR-TR-82-0874 NL

1 OF 1
AD A
120351

END
DATE
FILMED
11-82
DTIC

AFOSR-TR- 82-0874

(8)

AD A120351

THREE COMPUTER BASED AIDS TO MAINTENANCE SCHEDULING

AFOSR RESEARCH GRANT [redacted] AFOSR-81-0168

JUNE 30, 1981 - MARCH 23, 1982

FINAL REPORT

TO

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
AFSOR/NM
BOLLING AIR FORCE BASE, D.C. 20332

BY

SCHOOL OF ENGINEERING
TUSKEGEE INSTITUTE
TUSKEGEE INSTITUTE, ALABAMA 36088

SEPTEMBER 13, 1982

AN ASSESSMENT OF AIR FORCE MAINTENANCE SCHEDULING PROCEDURES

by

Andrew Hargrove

ABSTRACT

The complexity of the scheduling problem is examined including the large numbers of aircraft, the recurring changes of maintenance rules, and the conflict between a desired smooth flow of aircraft into maintenance and the desired flight procedures. Survey of typical schedulers reveals a need to allow a simple operation of a very complex system. Survey of available models reveals Decision Oriented Scheduling System (DOSS) as the most appropriate computer aid. Analysis of DOSS verifies its versatility but reveals the need to simplify it to the level of the average Air Force scheduler's expertise, to further adjust programming statements to a more natural "English-like" language and to allow for easier and faster programming adjustments due to unscheduled maintenance changes.

I. INTRODUCTION: Predictable aircraft maintenance is a task of large magnitude but one that is manageable and not complex. The addition of unpredictable of "unscheduled" maintenance requirements caused by aircraft malfunctions and aircraft emergencies complicate the problem. A further complication caused by operational requirements makes the task unmanageable unless some computer based aids are used.

Several computer based aids have been used for systems of similar magnitude and complexity: A very frequently used digital model is one that organizes all systems rules and components into a state variable matrix with further provisions for possible optimizing various requirements. Hargrove[1] has, using these methods modeled and simulated systems and their operations ranging from transportation to electrical power.

The maintenance scheduling problem differs from the previously designed models and simulations in that the rules change often and unpredictably. The assessment of maintenance scheduling described herein requires computer based modeling but models that are uniquely suited for Air Force scheduling solutions. DOSS (Decision Oriented Scheduling System) appears to have the greatest potential for satisfying the maintenance requirements.

II. OBJECTIVES OF THE RESEARCH EFFORT: Orginal objectives of the research effort were to:

a. Survey current Air Force scheduling methods and their rationale.

b. Study the problems actually faced by the scheduler in the field.

c. Analyze various digital scheduling models.

d. Suggest techniques that would improve the scheduling process.

The last objective was later expanded to recommend continued study of a computer based model (DOSS) that appeared to have the greatest potential for improving scheduling.

III. SCHEDULING RATIONALE AND METHODS:

a. The approach to surveying scheduling rationale and methods was to study major documents that relate to the topic. Official goals of major Air Force commands may differ somewhat. One universal goal, however, as described by Fallon[2], is to maintain mission ready aircrews and aircraft.

All commands operate decentralized operations wings at various locations, each of which conducts a flying program to maintain proficiency of aircrews and to exercise and maintain aircraft systems. Some of the aircrews and aircraft are also continually assigned to ground alert to serve as quick reaction forces.

Wing maintenance personnel inspect and prepare all aircraft for flight. Intermediate subgoals, say at wing level, are, therefore, alert, aircrew proficiency, and maintenance of reliable ready aircraft, and they must all be satisfied if the overall mission is to be successful.

The following discussion which aptly describes scheduling rationale was taken largely from Fallon's[2], "Rule-Based Modeling As An Analysis Tool: Implications for Resource Allocation Within the Strategic Air Command."

Because of the complexity of jointly considering aircrew and aircraft needs, the flying and alert activities of aircrews and aircraft are determined separately by two suborganizations of the wing, using guidelines from higher echelons. The operations organization schedules crew members for sorties (training flights), ground alert, leave, and ground training; its major focus is on certain intermediate subgoals of the major command, namely, aircrews proficiency and the personal needs of the aircrews. The maintenance organization schedules aircraft for sorties, inspections, alert, and repairs, its major focus is on different intermediate subgoals of the major command, namely, maintaining reliable ready aircraft and the personal needs of maintenance crews.

It is within the operations and maintenance suborganizations where operationally meaningful goals are developed that only indirectly relate to the overall official goal of the major command. These goals are reflected in scores of operational decision rules regarding the allocation of aircrews and aircraft for flying, maintenance, and alert activities. For example, the subgoal of aircrew proficiency is reflected by numerous rules for allocating aircrews to training sorties. The subgoal of maintaining reliable, ready aircraft is reflected by numerous rules for allocating aircraft to flying and inspection activities.

The development of rules, like most decision making in organizations, is largely adaptive in nature, with specific decision rules or standard operation procedures developed over time to deal with a variety of recurring situations.

Many of the decision rules, such as sortie requirements and inspection intervals, are actual stated policies, stipulated in regulations from higher headquarters. Many other rules, however, are local wing rules developed to implement major command policies and adhere to other subgoals within operations and maintenance. For example, a typical maintenance subgoals of having enough time to prepare the aircraft for alert, and ensuring the aircraft will be available to meet the maintenance alert schedule. These types of rules may vary across wings.

Although operations and maintenance schedulers focus on different parts of the wing resource-allocation problem, they are required, nevertheless, to coordinate their flying activities. Since they focus on different subgoals their preferred flying schedules rarely coincide. Through a process of negotiation and mutual adjustment a nomthly schedule is planned, and each week a firm schedule is determined. It is a formalized procedure resulting in an actual flying contract signed by wing executives.

As expressed above, the resource-allocation decision problem within wings is highly complex. It is a problem influenced by several decision makers and organizational units, and involves numerous organizational subgoals and constraints.

Futhermore, there is an inherent tradeoff relationship between numerous subgoals. The longer an aircraft is on the ground, either on alert or in an inspection, the less it is available for flying and the accomplishment of training sorties for aircrews. Moreover, the cause-and-effect relationships between alternative schedules and wing performance are not clear and involve a great deal of uncertainty.

Because of the complexity of resource allocation within wing, schedulers cannot consider all alternatives and pick the action with the best consequences. Instead, they find a course of action that is good enough; that satisfies. As mentioned earlier, a great many rules are developed in wing scheduling that stipulate what is meant by "satisfactory" achievement of a goal. Schedulers, for the most part, adhere to these rules and seek schedules that satisfy them.

Because of possible operations and maintenance emergencies, the enormous numbers of rules, and the vague nature of the goals, it is difficult enough to develop even one "satisfactory" schedule, much less search for "better" ones.

b. The result of the study is to verify and concur with the current Air Force scheduling plan, documentation and implementation in principle. The operational planning cycle including long range, monthly, weekly and daily plans are well documented in references[3 4] along with appropriate forms. The actual resoruce allocation problem of a wing, however, is a highly complex problem as previously described. Due to the enormous scale of the problem (number of resources, constraints and objectives), its dynamic nature (continually changing resources, constraints and objectives) and the difficulty of ever manually formulation these factors, a "rule based" computer system appears to be the most appropriate aid to scheduling. This computer aid will be discussed later.

IV. PROBLEMS OF THE SCHEDULER IN THE FIELD:

a. Problems of the scheduler in the field were surveyed in conversations with highly skilled staff personnel of the Air Force Logistics Management Center all of whom have had extensive experience in scheduling or maintenance, as well as schedulers and maintenance supervisors of an actual operational maintenance squadron. The problems appeared to be threefold. The volumnous amounts of data that must be handled, the ever changing rules and contraints of maintenance, and the degree of computer based skills needed to operate a rule-based or other digital scheduling system.

Jones[5] describes the multitude of variables that the scheduler must consider, including the fact that operational readiness is strongly influenced by flying. Unnecessary flying degrades readiness and ought to be curtailed. The scheduler is affected by both flight and maintenance work schedules. Personnel must be available to service aircraft problems at the end of a mission. It is completely unacceptable to allow ready aircraft to stand idle awaiting maintenance personnel. He has to consider alert and inspection requirements and availability of aircraft for maintenance training. He faces a high degree of uncertainty as to when components will fail and how long the repairs will take.

He must allow time between sorties for repairs, but must avoid excessive maintenance overtime and excessive "out of service" time for aircraft. In addition to all of the foregoing, the scheduler must play his important role in the operational planning cycle.

This section from Ledger and Peacock[6] contains the major scheduling rules and strategies employed at a representative organization within the Tactical Air Command in support of F4E aircraft maintenance scheduling. The information was obtained during interviews of aircraft maintenance scheduling personnel assigned to the representative organization. Major factors affecting rules and strategies of this organization are:

* Number of aircraft assigned -65
* Average number of aircraft possessed -64
* Average monthly flying hour allocation - 1156 hours
* Average monthly sorties launched - 1080
* Average sortie length -1.45 hours
* Aircraft are maintained under the phased inspection concept
* Phased inspection inteval - 100 hours
* Corrosion control (wash) interval is 30 days
* Average aircraft lost ot PDM (programmed Depot Maintenance) - 1 each 4 months
* PDM interval - 4 years
* The primary mission of the representative unit is student training

SCHEDULING PROCESS. The representative unit is required to maintain a quarterly forecast of maintenance requirements. The displays for the third and second month reflect minimal useful information and are not published for general use. The third and second month schedules are accomplished each month's schedule. These displays consist of data extracted from the operations quarterly forecast, the total number of flying hours required per month, and the total number of phase inspections to be generated for each month. The PDM, aircraft transfers and special requirements, such as the "Weapons System Evaluation Program" (WSEP), provide the known aircraft utilization and are posted as film dates. Included in production of the third and second month's schedule is the 90 day forecast of time change requirements. This serves as notification to supply of the requirement for replacement parts. Other considerations are: anticipated facility/equipment shortages, i.e., eighty percent of the radar systems specialists are currently at the apprentice level. The significant scheduling actions take place during development of the monthly and weekly schedule.

CURRENT MONTH SCHEDULE. The activities displayed on the monthly schedule are those activities which have known or estimated due dates occuring during the applicable month. A "Log Book" is maintained which reflects the "due date", "completed date". and "scheduled date" for inspections which are accomplished on a calendar basis. The Documentation Section provides a listing of all items due during the applicable month. The monthly schedule is developed on an AF Form 2401 worksheet. The firm PDM, transfers, and special activities are posted to the worksheet along with the activities extracted from the "log book". Phases and related pre-dock dates are developed and posted to the worksheet. The worksheet is reviewed for balance and flow. If any overall activity reveals "peaks and valleys", the schedule is adjusted to evenly distribute that activity for the affected aircraft over the complete month. When all known requirements have been posted to the worksheet and final adjustments are made, it is published as the monthly "Equipment and Utilization Maintenance Schedule" and becomes the monthly reference, providing inputs to development of the weekly schedule.

WEEKLY SCHEDULE. Development of the weekly schedule is initiated on Thursday each week, and published on Friday for the upcoming week. The weekly "work up" is accomplished using a worksheet. The method of posting is an iterative process of systematically recording groups of activities in order of the least adjustable to the most adjustable. After each iteration, the schedule is adjusted to eliminate activity conflicts and equalize activity distribution throughout the week. Considerations for input data to the weekly schedule, current are activities previously scheduled in the monthly schedule, current and projected aircraft status, systems capability measurements, aircraft delayed discrepancy files, and shop chief requests. Specific mission requirements are a product of the weekly operations/maintenance scheduling meeting. Adjustments occur during second or subsequent iterations of the schedule when review of the aircraft and individual systems status reflects an impact on the related activity. For example, on the first interation an aircraft was posted for a due, radar calibration inspection. Review of the air craft status reveals that the radar set is inoperative and is not estimated to be in operation by the scheduled due date. The aircraft radar calibration check would be deferred to the following weeks schedule or scheduled for a different day during the week based on parts. The aircraft would become available for non-redar missions, PM, USM, (unscheduled maintenance) or a down day. After all non-flying activities have been adjusted, all remaining aircraft are considered to be potential "flyers" . These potential flyers are accumulated by row assignment. The goal is to have an equal number of aircraft per row, per day, available for flying. (Under conditions existing at the time of this report, the minimum was seven per row.) If an imbalance exists, the schedule is again reviewed. Weapons load training, training support, and down days can be moved from one row to another. If an equal number of aircraft per row cannot be achieved, the scheduled dates of like activities any be exchanged between aircraft of different rows. These adjustments usually will accomplish the desired balance. Flying activities are scheduled using the available aircraft from each row. The primary consideration at this point is "time remaining to the next phase inspection". The objective is to generate an aircraft utilization rate to maintain the predetermined phase input flow. After aircraft availability by row has been established, aircraft selection for missions/sorties is determined by application of established policy and mission requirements.

The least adjustable activity is the phase inspection. To clarify, once the desired utilization rate has been determined and the estimated due date has been established, then the phase becomes the least adjustable. Categorically, the most frequently recurring activities, i.e./ "Phase" and "Wash/Corrosion" are the least adjustable and the least frequently recurring or "one time" activities are the most adjustable. Activities whose duration is short enough to allow flying on the same day are also highly adjustable.

SCHEDULING GOALS.

Goal. Enhance morale among maintenance personnel.

Strategy. Distribute flying activities equally throughout the maintenance organizational structure. Allocate an equal number of aircraft from each row to the flying schedule. This provides an equal number of preflight, launch, and recovery actions in each OMS section. It allows each row supervisor to better predict requirements and obtain maximum utilization of his assigned personnel.

GOAL. Equalize phase dock loading and enhance specialist morale. Maintain equal distribution of phase time on all aircraft. Minimize NORM time accumulation during phase inspection.

STRATEGY. Establish and maintain balanced phase inputs. Estimate phase due dates based on equitable utilization rates. Control aircraft flying activities to insure phase input dates are met. Input one aircraft on Monday, one on Tuesday, and one on Wednesday, and complete all phase inspections before each weekend. Adjust phase inputs as required to prevent NORM time accumulation on holidays. Solicit schedule inputs from shop chiefs and coordinate "high specialist use requirements" with the affected shops.

ACTIVITY DESCRIPTIONS. The following activity descriptions include statements representing scheduling goals and policies. Resources required for completion of the activity are described by name and characteristics. Due to the nature of this study, more emphasis has been placed upon describing attributes of aircraft than attributes of other resources.

Constraint rules utilized to control the availability of resources are provided as statements indicating the factors that determine non-availability of resource.

Ranking represents the resource attributes that the scheduler can arrange in the order of importance to satisfy current requirements. These are not presented in any specific order since the relative "weight" assigned to selected attrivutes may by altered to meet specific mission requirements.

ACTIVITY: PHASE

- Phase inspections are required at 100 hour intervals.

- Phase inputs are forecast for a 90 day period. A 30 day forecast is made each month. Minimum chages are made to the 30 day forecast since required "time change" items are requisitioned based on the input forecast.

- Phases are numbered 1 thru 6.

- Maximum time remaining at input is 20 hours.

- Desired time remaining at input is five hours or less.

- Aircraft cannot be flown if the inspection is due.

- Phase pre-dock meeting is conducted a minimum of 3 days prior to phase input. Pre-dock may be conducted as much as 5 days prior to input.

- The phased inspection nearest to 1200 hours of engine operation is considered an "engine change phase". Aircraft that require an engine time change are input to "engine change" the week prior to phase input. This aircraft is always input to phase on the Monday following the engine change. (Engine change will be complete and the aircraft will be "OR" prior to Phase input.)

- Engine changes are scheduled for input a minimum of three days prior to phase input.

- Avoid scheduling two "engine change phases" during the same week.

- Six dock crews work a five day week. When not allocated to a phased inspection, the crew is used to augment the preventive maintenance crew.

- Normal inspection thru put is three days. Normal input days are Monday, Tuesday, and Wednesday; output is scheduled for Wednesday, Thursday, and Friday, respectively.

- On week when four phased inspections must be accomplished an additional aircraft is input on Thursday. Output is scheduled for Saturday.

- Inspection crews providing coverage on Thursday inputs will work Thursday, Friday, and Saturday. They will not be available on Monday. (This may affect preventive maintenance scheduling.)

CONSTRAINTS:

- If undergoing extensive maintenance the aircraft is not available.
- If time to phase is greater than 20 hours, the aircraft is not available.
- If due an engine change, the aircraft is only available for input on Monday following the engine change.

The long range planning cycle which displays as a minimum the current month and the following two months maintenance requirements starts the planning cycle.[3] Monthly, weekly, and daily plans are made and new items added as received until all known maintenance requirements are posted. Computer based aids are now available but may require skills not possessed by the average scheduler.

b. Need for a "Rule Based" computer. To accept, collate, and refine the large amounts of maintenance data, as well as accepting and applying rele changes by emergencies and adjustments, a suitable, perhaps "rule based", computer is required.

To assure that these digital systems and programs can be used by the average airman, they must be designed or redesigned so that ordinary English or English-like languages may be used by schedulers.

V. ANALYSIS OF VARIOUS COMPUTER AIDED METHODS:

a. PLANS, PLUS and TWS

(1) PLANS is a computer language designed especially for scheduling, plus is a collection of canned programs thatused the PLANS language, and TWS is a software tool intended to develop translators for programming languages including PLANS and PLUS. Their descriptions below are quoted directly from their defining documents.

PLANS[7], PLUS[8], and TWS[9] are associated with the development of a scheduling language sponsored by the NASA Lyndon B. Johnson Space Center. The work began in 1973 in response to a need to reduce the cost of developing and maintaining software to support scheduling and resource allocation tasks. Three related products were developed. The first, callsed PLANS, is a programming language that is ideally but not uniquely suited to writing scheduling programs. The second, called PLUS, is a library of utility programs representing logic that is common to a broad range of operations planning and analysis software. The third, called TWS, the Translator-Writing System, was used to develop and maintain the PLANS language translator, but has general capabilities well beyond its role in the PLANS system.

(2) The design of the Programming Language for Allocation and Network Scheduling (PLANS) was prompted by the inadequacies of existing languages being used to solve scheduling problems. A high-level language was needed that would allow easy, direct expression of the kinds of functions frequently found in scheduling any resource allocation programs. PLANS fulfills this need primarily because of its unique capability to allow dynamic manipulation of tree data structures at execution time. Another important feature is the close correspondence that exists between basic scheduling functional operations on the one hand and PLANS statements on the other. This allows both the initial programmer and the maintenance programmer to design and modify PLANS programs easily. These powerful language fetures make it applicable to many areas other than scheduling. That is, PLANS is not a special purpose scheduling language, even though it was motivated by scheduling problems. It is <u>generalized, high-level tree manipulation language</u>.

BACKGROUND OF PLANS

Although its capabilities have proven to be much more broadly applicable, PLANS was designed to acheive a single goal:

> to allow the designer of experimental or constantly changing scheduling and resource allocation algorithms to translate his algorithm designes to working code directly from their basic functional descriptions, without intermediate detailed program design steps, without highly specialized programming expertise, and at minimum span time and manpower costs.

The necessity to go through several additional design and implementation steps before the advent of PLANS resulted in unacceptably long development times and high costs. Equally important, it tended to discourage the truly experimental approach to scheduling algorithm development which holds the greatest promise of vonvergence on good solutions for large, logically complex scheduling problems. PLANS was designed, then, to cut development cost and span time, and also to provide a medium for easy modification of scheduling program.

An analysis of previously existing programming languages as applied to scheduling problems revealed two deficiencies: (1) the language level did not correspond to the level of the functions typical in scheduling algorithms, and (2) more significantly, the data structures (usually only arrays) of the languages did not correspond to those typical of scheduling problems.

Scheduling problems typically involve information structures which are logically hierarchical. A schedule consists of jobs, each of which has certain properties of its own (time of occurrence, duration, name, etc.), and each of which also has certain relations to other jobs (predecessors, etc.) and to particular resources which to perform the jobs. These resource assignments have, in turn, such properties as time of occurrence, duration, etc. The inputs to scheduling algorithms are also typically hierarchic in nature, involving, for example, information about resources, which breaks down into resource types, each of which in turn may involve many resource units, each of which has its own physical and logical properties (weight, location, etc.), and each of which is also unavailable at certain times due to prior assignments to jobs. The necessity to represent information of this sort in the form of arrays (as when programming in FORTRAN, for example) led to programs which were quite large, difficult, and unreadable. This is due to the overwhelming preponderance of indexing operations and similar functions required to express, in array form, information which is not logically of an array character.

As a result of these considerations, PLANS is designed around a single feature which is unique among high-level languages: the provision of hierarchic data structures--trees-- whose structure, as well as data content, can be manipulated at execution time. Many languages (e.g., COBOL, PL/1, ALGOL) have hierarchic data structures which are static during execution. The feature of PLANS which is novel (except, perhaps, among difficult-to-use list-processing languages), is its dynamic manipulation of trees. The output usually required of scheduling programs is, in large part, a restructuring of the input, which can be most easily accomplished in a language which allows direct restructuring of its data structures.

(3) PLUS is the acronym for "Program Library of Utilities for Scheduling". PLUS contains a collection of applications programs called modules, each of which is designed to provide a portion (or all) of the logic needed to construct a computer program to solve problems associated with scheduling and/or resource allocation.

The PLUS modules are not constrained to any particular application. They are methodological segments that may be used on any problem for which the corresponding methodology applies. The methodologies associated with the PLUS modules are those commonly, but not exclusively, used in solving scheduling and resource allocation problems. For example, sorting, ordering, interval algebra and set operations are functions common in scheduling algorithms. These functions (and many others) reside in PLUS. They are equally applicable to full range of operations analysis problems.

The PLUS modules are an integral part of a programming system that is designed to reduce the labor and time needed to design, develop, revise and maintain computer programs associated with a very general class of problems. The other elements of this system are a general descriptive framework for describing problem data, and a special high-level programming language. The descriptive framework takes the specific form of a set of standard data structures that can, at the user's option, be used to capture solution information as data for PLUS modules or other computer programs. The standard data structures are the formats assumed and generated by the PLUS modules. Thus, the use of PLUS is facilitated if the user describes problems in terms of the standard data structures. The standard data structures assumed by the PLUS modules are particular examples of the data structures associated with the special programming languages called PLANS. The relationship between PLUS and PLANS is described briefly in the following section.

RELATIONSHIP OF PLUS TO PLANS

The original source code for the PLUS module is written in a programming language called PLANS (Programming Language for Allocation and Network Scheduling). PLANS is designed around a single feature which is unique among high-level languages: the provision of hierarchic data structure--trees--where structure, as well as data content, can be manipulated at execution time. The standard data structures that are assumed as inputs to the PLUS modules and which are produced as outputs by many of the PLUS modules are specific examples of PLANS tree structures. Therefore, the PLUS modules are designed to be called from programs written in PLANS. The PLUS modules that do nor tequire or produce standard data structures will require or produce structures that are PLANS trees. Thus, the use of PLUS is (for all practical purposes) limited to programs written in PLANS.

(4) TWS is the acronym for Translator-Writing System. TWS is a powerful software development tool which has numerous areas of application, but which is especially intended for use in the development of translators for high-level computer programming languages. The system accepts as its input a definition of the translation in the form of a syntactic definition of the source language, with embedded semantic information. The output of the system is a working translators. Translators developed via the TWS tend to be more flexible and more easily modifiable than those developed through manual methods, and are more rapidly developed. Because of the way in which the translator's function is specified, TWS translators tend also to be quite rigorously defined; this may result in greater reliability or freedom from errors.

There are two manual inputs which are mandatory: the formal language definition and the token definition. These operations are defined in detail by Ramsey[9]. Briefly, the language definition is a formal description of the mapping from the source language to the desired object language. This mapping is expressed in the form of a grammar (syntactic defintion) of the source language which contains embedded semantic information expressed in terms of the object language. The source language is thus defined in terms of the object language. The source language is thus defined in terms of the object language. Such a grammar, augmented by semantic information, will be referred to as an "augmented grammar".

The token definition is a description of the basic elements of the source language in terms of the characters which comprise them. This input contains definitions of the formats of identifiers, numberic constants, etc. The definition takes the form of a state transition matrix. Occasionally, it may be necessary to modify the lexical analyzer (subroutine) itself, but a tabular input is sufficient for most stream-oriented source languages.

(5) PLUS with its supporting PLANS and TWS is an excellent and versatile scheduling aid. The level of expertise required for implementation however is much too involved for any scheduler. Nor does there appear to be the likelihood of it being reduced to a workable levels.

b. MMICS (Maintenance Management Information and Control System). MMICS[10] is a computer-assessed management and control system designed to improve the effectiveness and efficiency of base level maintenance. It does not schedule, but accumulates and categorizes necessary data so that manual scheduling is easier. It prints operational schedules, relieves personnel from the task of recording and maintaining operational information, monitors starts and stops of operations, and provides data that can be used to analyze operations.

It is an exceptional management information system.[11] It maintains historical data on aircraft and personnel, it answers "real time" inquiries, it prepares needed reports, and it performs each of the basic computations that the scheduler requires.

Yet, it only provides information to the scheduler, it does not actually participate with the scheduler in the decision making process. It does not assist in the selection of the optimum design of the schedule or in the selection of the most appropriate tail numbers to fill the schedule. These functions are performed through a manual process.

c. SASS (Standard Automated Sortie Schedule). SASS[12] is a computer program used for scheduling aircraft sorties. This system schedules aircraft in such a manner that maximum flying hours are realized and a smooth progression into scheduled inspection is maintained. Several types of input are loaded into the computer. Each individual aircraft tail number is assigned a status code as to preference for flight status. Code one means "fly first", two means "fly according to smooth flow" and three means "fly last if needed". The daily sortie schedules, as well as the number of spare aircraft, are also fed into the computer. After these and other data are loaded, the daily schedule is produced by the computer. Maintenance must insure there is enough flyable aircraft to meet the schedule.

d. DOSS (Decision Oriented Scheduling System). DOSS, with the assistance of the scheduler and using a "rule based" language, not only completely schedules aircraft operations and maintenance, but allows for impromptu changes, additions, and deletions of requirements.[13]

By "rule based" languages we mean the schedulers' describe problems not in programming language but in a more natural "English-like" language. This is one of two unique features of DOSS. The other is there are no fixed rules in the system. Instead, the algorithm allows rules to be created, added, deleted, or changes at will.

The scheduler does not push buttons, but instead provides continuous guidance in the form of rules, goal tradeoffs, and decisions.

DOSS functions at wing level and receives actual wing data and current rules. The rules may be constraints, perference or processes.

The actual monthly or weekly schedule is printed or shown on video or in graphical form with numbers and symbols thereon for ease of evaluation by the scheduler. If the schedule is not acceptable, the scheduler may alter, delete, or add new rules to the process to correct the schedule.

VII. RECOMMENDATIONS:

In today's Air Force, where both operations and maintenance have evolved into complex and sophisticated systems, effective scheduling is an absolute necessity. The system now employed is an excellent one in principle. It does serve the goals of maintaining mission ready aircrews and aircraft.

The magnitude, complexity and variations in scheduling, however, have grown to such proportions as to require some computer aided assistance. Of the several computer aids studied, it appears that the Decision Oriented Scheduling System (DOSS) is by far the best and has the greatest potential.

Recent research revealed a need to implement, test, and further analyze DOSS and its applications to maintenance scheduling. The study just completed had as its only shortcoming insufficient program testing and analysis due to the short duration of the project.

It is recommended that further research with DOSS continue with emphasis on actual digital computation. Possibly, simplification of the DOSS rule-based language to the level of ordinary airmen is also recommended as a subsequent goal.

SUMMARY OF DOSS STUDY

James E. Cooper is a senior Electrical engineering student at Tuskegee Institute. He has had a wide variety of computer programming and operating experiences and has acted as a manager of the Tuskegee Institute Computer Center. He acted as assistant principle investigator for this research project. The following are his summary and recommendations.

The disadvantages found in using DOSS were:

(1) Due to the level of difficulty of the DOSS package, Air Force personnel with considerable experience in computer usage must be used as schedulers.

(2) It is easy to inadvertently exit from the DOSS environment. Thus the possibility of losing recently input scheduling information exists.

(3) The system does not have the ability to easily and quickly accept and implement scheduling changes.

The single greatest disadvantage of the DOSS package is the lack of Air Force personnel with enough experience with DOSS to fully utilize the system. The DOSS package is written in several programming languages, thus requiring personnel with experience in these several languages and who can therefore trouble shoot the package if the need arises. It is recommended that DOSS either be converted to a single language program or that a scheduling package be developed using a single language.

The DOSS package should be more of a "menu" driven package in which the user is prompted for input from the terminal and has the option of getting programming help from DOSS during any stage of the input session.

REFERENCES

1. Hargrove, Andrew, "A Non Linear Constrained Multivariable Digital Controller that is Insensitive to the Initial State". Proceedings of the IEEE Region 3 Annual Conference, New York, N. Y., 1976.

2. Fallon, Richard, "Rule-Based Modeling as An Analysis Tool: Implications for Resource Allocation Within the Strategic Air Command", RAND Note N-1489-AF, Rand Corporation, Santa Monica, CA 1980.

3. Air Force Manual 66-1. Maintenance Management, Vol. II, ATC Supplement 1. Randolph AFB, TX: HQ ATC, 22 August 1975.

4. Exentsion Course Institute, "Career Development Course 39270, Maintenance Management Technician, Volume 5, Planning and Scheduling" 3350 Technical Training Group, USAF School of Applied Aerospace Sciences, Chanute Air Force Base, Illinois, 1976.

5. Jones, Larry P., "An Analysis of Aircraft Scheduling Methods for Mutiple Sortie Aircraft", Unpublished, Air Command and Staff College, Maxwell AFB, AL, May 1976.

6. Ledger, C. R. and Peacock, W. L. Scheduling Rules Study, Logistics Data Collection Report, Air Force Data Systems Design Center, pp TAC2-2 - TAC 2-19, Potomac Research Inc., 1975.

7. Ramsey, Rudy H., Willoughby, John K., and Kullas, Daniel A. A User's Guide to the Programming Language for Allocation and Network Scheduling (PLANS). Technical Report SAI-77-068-DEN, Science Applications Inc., Englewood, CO. 1977.

8. Kullas, Daniel A. and Willoughby, John K., A User's Guide to the Program Library of Utilities for Scheduling (PLUS). Technical Report SAI-77-069-DEN, Science Application Inc., Englewood, CO. 1977.

9. Ramsey, Rudy H. A User's Guide to the Translator-Writing System (TWS). Technical Report SAI-77-067-DEN, Science Applications Inc., Englewood, Co. 1977.

10. "Maintenance Management Information and Control System (MMICS)." TIG Brief , Bol. 26, No. 21 (8 November 1974).

11. Air Force Manual 66-278. Maintenance Management Information and Control System, Vol. IV. Washington: Department of the Air Force, 1 April 1975.

12. Air Training Command Manual 66-422. Standard Automated Sortie Schedule (SASS). Randolph AFB, TX: HQ AT, 1 July 1975.

13. Hillestad, Richard J., "Review of the DOSS Research: A Briefing". (Report WN-9771-AF), RAND Corporation, Santa Monica, CA April 1977.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 82-0874	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THREE COMPUTER BASED AIDS TO MAINTENANCE SCHEDULING,		5. TYPE OF REPORT & PERIOD COVERED FINAL, 30 JUN 81-23 MAR 82
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Andrew Hargrove		8. CONTRACT OR GRANT NUMBER(s) AFOSR-81-0168
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Electrical Engineering Tuskegee Institute Tuskegee Institute AL 36088		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE61102F; 2304/A6
11. CONTROLLING OFFICE NAME AND ADDRESS Directorate of Mathematical & Information Sciences Air Force Office of Scientific Research Bolling AFB DC 20332		12. REPORT DATE September 13, 1982
		13. NUMBER OF PAGES 22
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer based maintenance aids.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The complexity of the scheduling problem is examined including the large numbers of aircraft, the recurring changes of maintenance rules, and the conflict between a desired smooth flow of aircraft into maintenance and the desired flight procedures. Survey of typical schedulers reveals a need to allow a simple operation of a very complex system. Survey of available models reveals Decision Oriented Scheduling System (DOSS) as the most appropriate computer aid. Analysis of DOSS verifies its versatility but reveals the need to simplify it to the level of the average Air Force scheduler's expertise, to further adjust (CONTINUED)

DD FORM 1 JAN 73 1473

82 10 12 172

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ITEM #20, CONTINUED: programming statements to a more natural 'English-like' language and to allow for easier and faster programming adjustments due to unscheduled maintenance changes.

LME

-8